

JUMP to .NET

Java Users Migration Path to .NET

Tony Goodhew
Product Manager
Microsoft Corporation



JUMP to .NET

Goals

- 1. Provide Developers the Java language on the .NET Platform**
- 2. Seamless upgrade to the .NET platform for Visual J++ Developers**

What we aren't doing

- ◆ **We are not building a Java platform**
 - ❖ **The aim of the product is not to support development against the Java platform on .NET**
- ◆ **We are not following random class introduction from Sun**
- ◆ **We are not implementing J2EE**

The Java Language

Where JUMP fits in

C++

**C++
Language**

**C++ Run
Time Library**

OS Platform

Java

JUMP

**Java
Platform**

Why .NET?

- ◆ **Preserves existing investments**
 - ❖ **Access the value of the Microsoft platform**
- ◆ **Enables timely and proactive updates to Microsoft developers**
 - ❖ **Execution Environment**
 - ❖ **Programming Tools**
- ◆ **Exposure to .NET**
 - ❖ **Create and Consume Web Services etc**
 - ❖ **Cross language development**

JUMP to .NET

Developer Choice

Path 1: Use the Java language on .NET

What Microsoft is delivering

Tools

- Java-to-MSIL translator
- Binary component converter
 - Convert .class files to MSIL
 - Can reuse as is
 - Can convert to MSIL
- Interop layer - for library support at JDK1.1.4 level
- Visual Studio.NET integration
 - Java as a 1st class language
- Financial assistance with your

What this means for developers

Options

- Re-build existing VJ++ source to run on .NET CLR
- Continue writing new code in Java
- Can use Java.* class packages as defined by JDK 1.1.4
- Regarding existing Java .class files
 - Can reuse as is
 - Can convert to MSIL

Path 2: Full Migration to .NET

Tools

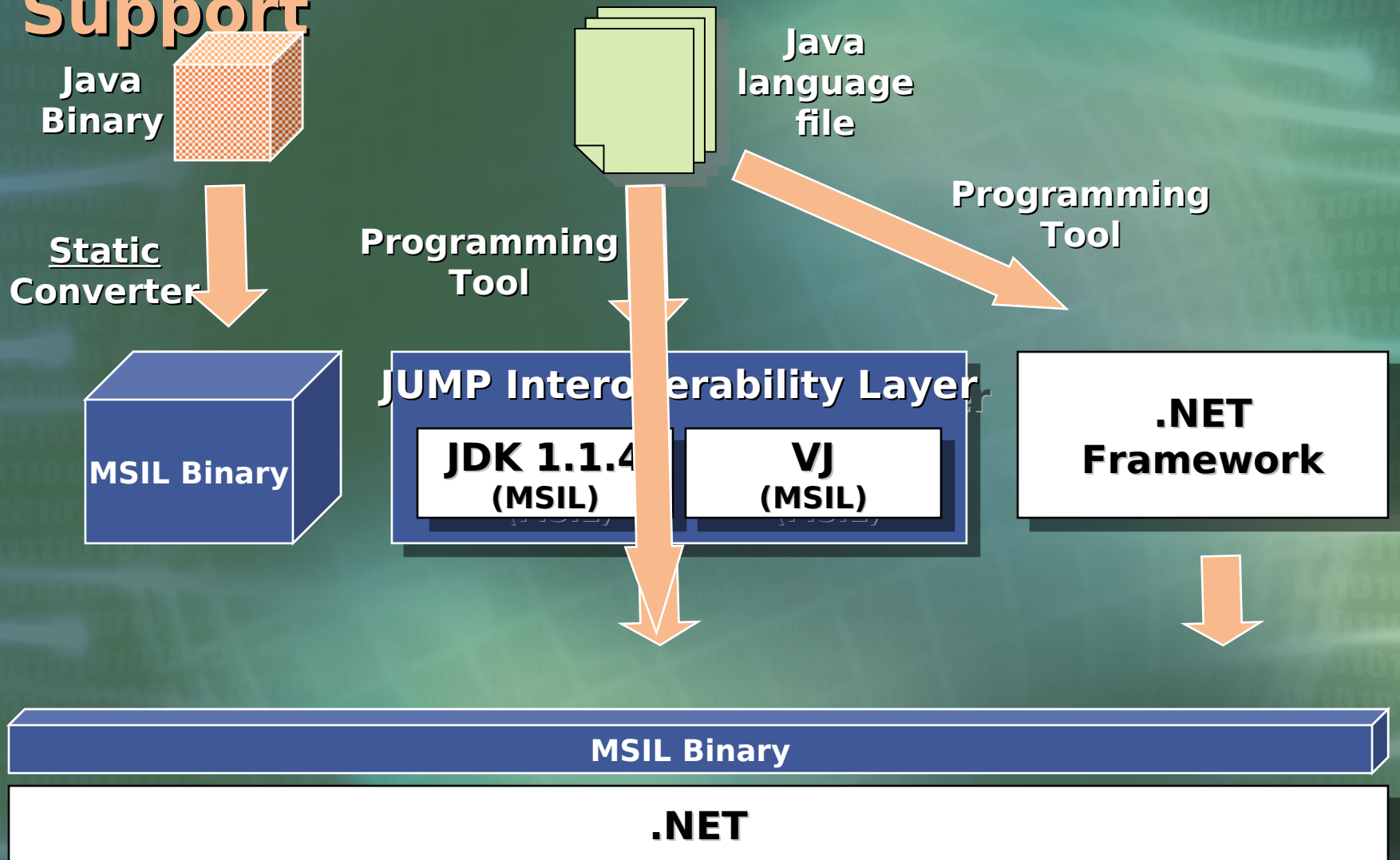
- Java-source-to-C# source translator/converter

Options

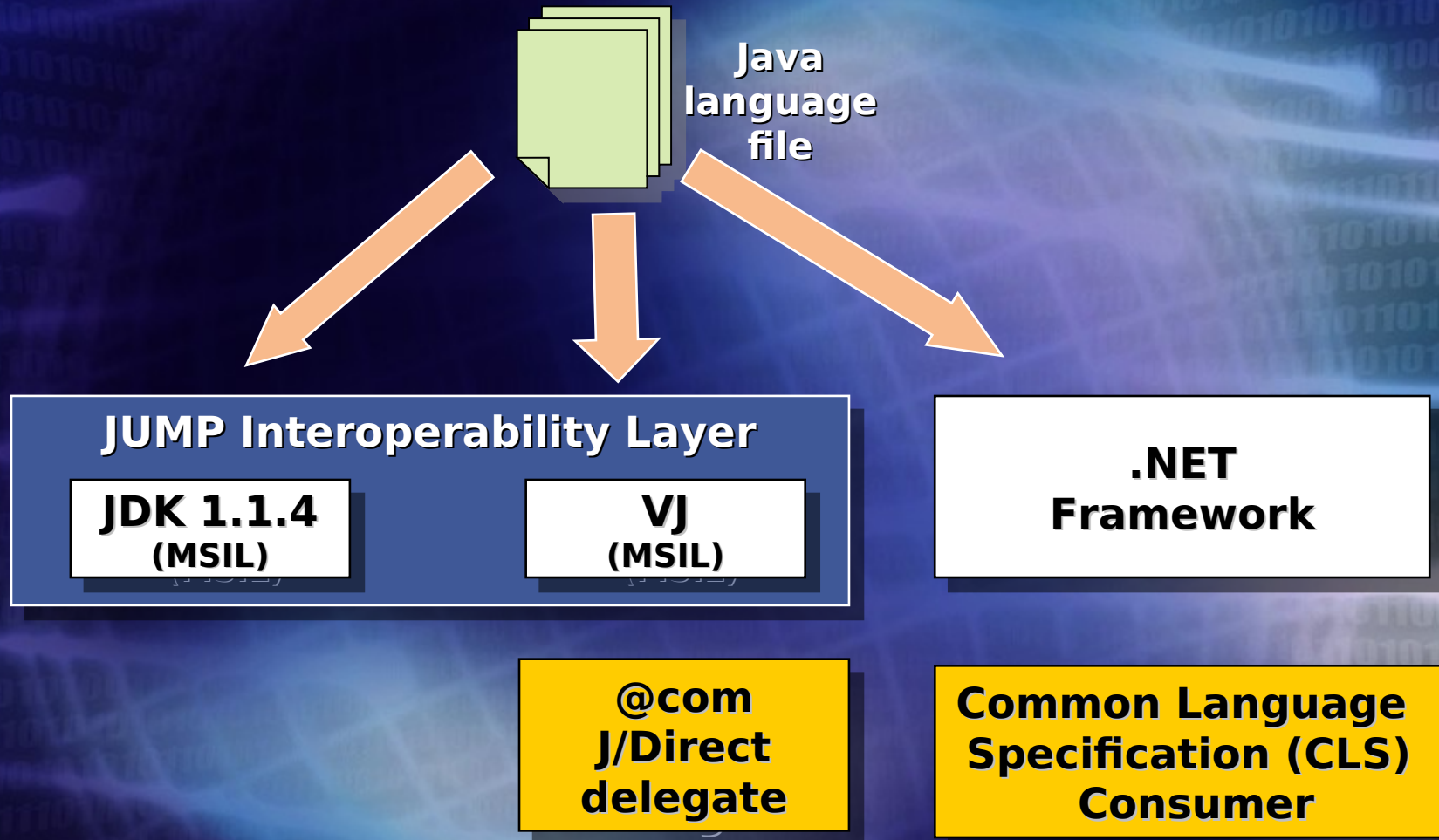
- Convert existing Java modules to C#
- New development in C#
- Projects can reference Java classes running on the interop layer (from above)

JUMP to .NET

Path 1 - Interoperability + Tools Support



Java Language Issues



Library Relationship

.NET Framework

System.Object

...

...

...

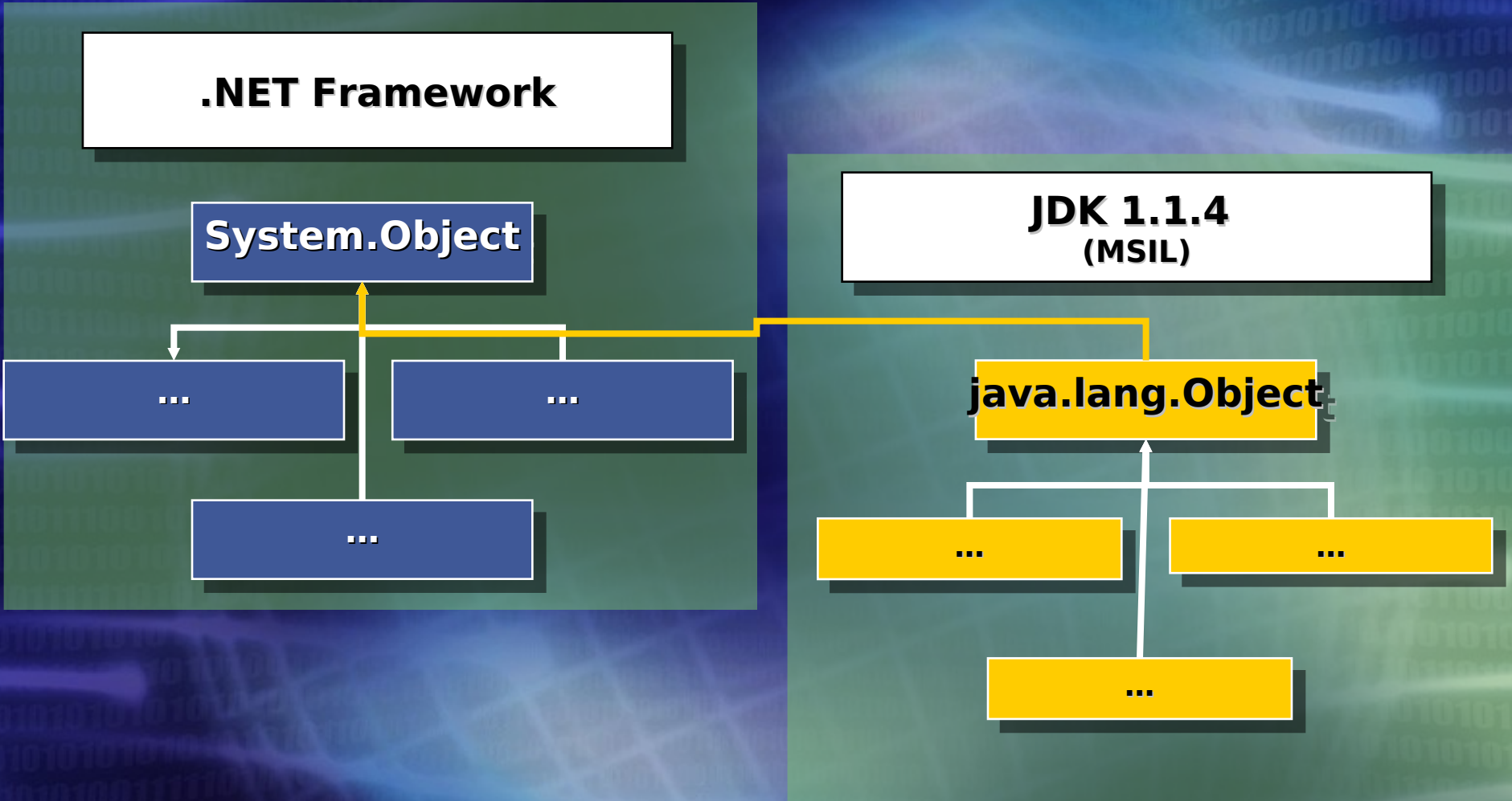
**JDK 1.1.4
(MSIL)**

java.lang.Object

...

...

...



Beta 1 product features

◆ IDE Integration

◆ JDK 1.1.4

◆ Javabeans

◆ J/Direct

◆ Partial support for com.ms.*

JUMP to .NET

Developer Choice

Path 1: Irrational refusal to decommit from Java syntax

What Microsoft is delivering

Tools

- Java-to-MSIL translator
- Binary component converter
 - Convert .class files to MSIL
 - Can reuse as is
 - Can convert to MSIL
- Interop layer - for library support at JDK1.1.4 level
- Visual Studio.NET integration
 - Java as a 1st class language
- Financial assistance with your

What this means for developers

Options

- Re-build existing VJ++ source to run on .NET CLR
- Continue writing new code in Java
- Can use Java.* class packages as defined by JDK 1.1.4
- Regarding existing Java .class files
 - Can reuse as is
 - Can convert to MSIL

Path 2: Full Migration to .NET

Tools

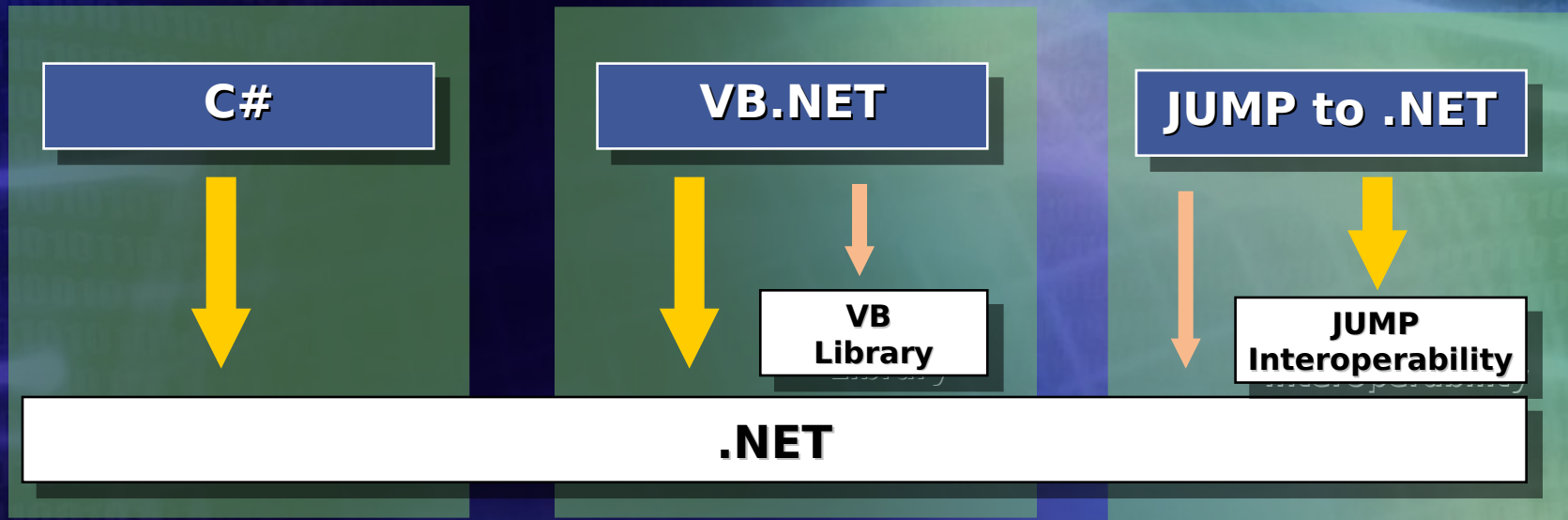
- Java-source-to-C# source translator/converter

Options

- Convert existing Java modules to C#
- New development in C#
- Projects can reference Java classes running on the interop layer (from above)

Migration to C#

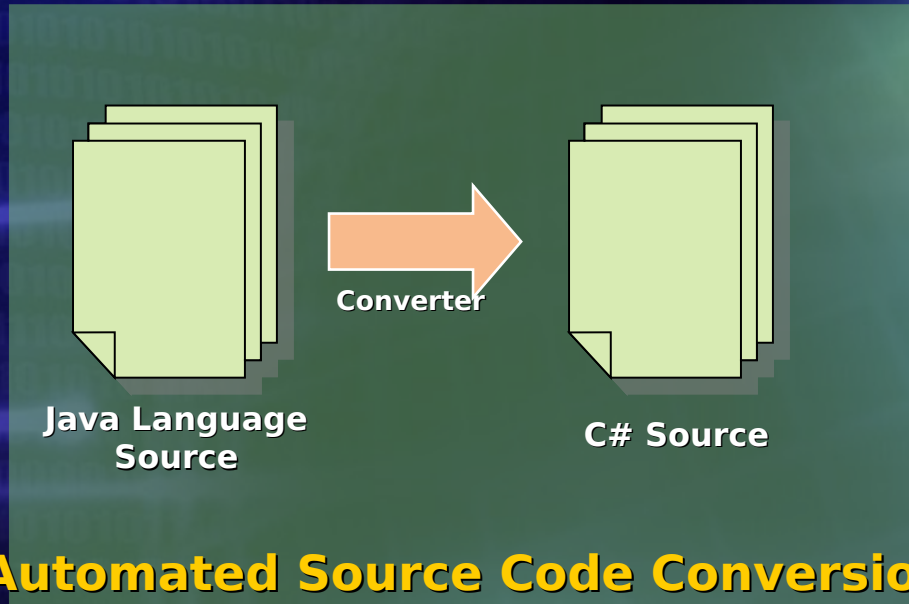
Path 2 - Changing Language



- ◆ **Adopt .NET**
 - ❖ **Want full fidelity with .NET Framework**
 - ❖ **No marshalling between libraries required**

JUMP to .NET

Conversion and Migration









Beta 1 Product

Features

- ◆ Declarations
- ◆ Source Code Conversion Tool
- ◆ Array Access
- ◆ Assignment
- ◆ New
- ◆ Method invocations
- ◆ Array Initializers
- ◆ Type conversions
- ◆ Many control structures (Loop, Switch, For etc)
- ◆ Exception Handling
- ◆ Delegates
- ◆ All WFC Classes
- ◆ Most of the core java packages.

Customer Flexibility

- ◆ Multiple options to preserve, enhance and migrate projects to .NET Platform
- ◆ These options can be blended

	Java Language		.NET Platform	
	Source	Binary	Source	Binary
Programming Tool				
Binary Conversion				
Source Code Conversion				

Example 1:

Visual J++ Customer with WFC for GUI apps

- ◆ 200 different classes developed
- ◆ Heavy use of WFC; little or no JFC/Swing
- ◆ Recommended course of action:
 - ❖ Now: Continue development with VJ++
 - ❖ In 3 months: Join Beta 2 and hit F5
 - ❖ Long term: Extend application using Visual Studio.NET

Example 2:

J2SE with JFC for client-side GUI apps

- ◆ **20 different cross-platform applications developed**
- ◆ **Limited use of RMI**
- ◆ **Recommended course of action:**
 - ❖ **Now:**
 - ❖ **continue using Java for Cross-platform clients**
 - ❖ **Consider Java/SOAP toolkits for heterogeneous interop**
 - ❖ **In 3 months: Visual JUMP Beta 2 non-UI**

Example 3:

J2SE server-side apps, sockets for communications, fat Java client applications

- ◆ Limited GUI code; limited WFC, JFC, or Swing.
- ◆ Some use of RMI
- ◆ Recommended course of action:
 - ❖ Now: Develop generic Java
 - ❖ In 6 months: Use Beta 1 of Conversion Tools
 - ❖ Long term: Code in C# for .NET and use Web Services and Windows Forms

Example 4

Customer using J2EE

- ◆ **No GUI code**
- ◆ **Use of EJB Entity Beans and Session Beans**
- ◆ **Servlet and JSP for presentation layer**
- ◆ **Recommended course of action:**
 - ❖ **Now:**
 - ❖ **Examine costs of your existing Server software and hardware; measure existing developer productivity.**
 - ❖ **Evaluate Windows Server as your strategic server-side platform - meeting reliability, scalability, and agility requirements**
 - ❖ **Pilot C# and .NET - can convert some existing Java business objects.**
 - ❖ **Consider ASP.NET and ADO.NET as higher-performance options.**
 - ❖ **In 3 months: Make code generic by removing AppServer Code**
 - ❖ **Long term: Use Visual JUMP and .NET or Conversion Tools**

Common Questions

and the answers

- ◆ **What about JSP and Servlet? Will the JUMP tools be able to convert these to ASP.NET?**
 - ❖ **We are producing a conversion tool separately to the JUMP product**
- ◆ **Same question for JDBC code - will the JUMP source converter handle this?**
 - ❖ **JDBC is fully supported**
- ◆ **Will Microsoft be supporting JDK 1.3?**
 - ❖ **We are not building a Java platform or following random releases from**

Availability

- ◆ Expect tools beta 1H '01, final 2H '01
 - ❖ Signup to the beta by visiting:
<http://msdn.microsoft.com/visualj/jump/default.asp>
- ◆ Pricing announced later
- ◆ Pricing and availability of migration services announced later

Where do **you** want to go today?

Microsoft®